

## LAB 5

### MỤC TIÊU

Kết thúc bài thực hành sinh viên có khả năng:

- ✓ Biết cách sử dụng, custom Spinner và ListView.
- ✓ Biết cách tương tác với Spinner và ListView.

### NỘI DUNG

**BÀI 1: Sử dụng lại Lab3 tạo thêm activity AddStudent có giao diện như sau:**



Trước khi click vào Spinner



Sau khi click vào Spinner

## Hướng dẫn:

- ❖ Custom EditText: tạo file **custom\_edittext.xml** trong thư mục **drawable**, sau đó sử dụng file này làm **background** cho **widget** cần sử dụng.

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="rectangle">
    <stroke
        android:width="2dp"
        android:color="@color/black" />
    <gradient
        android:endColor="#FF9E80"
        android:startColor="#EA80FC" />
    <corners android:radius="20dp" />
</shape>
```

- ❖ Tạo widget **Spinner**, đặt id và ánh xạ
- ❖ Tạo class (**Model**) có tên **School**, có các thuộc tính sau:

```
public class School {
    3 usages
    private int image;
    3 usages
    private String name;

    5 usages  ± Dinh Nguyen
    public School(int image, String name) {
        this.image = image;
        this.name = name;
    }

    1 usage  new *
    public int getImage() { return image; }

    new *
    public void setImage(int image) { this.image = image; }

    new *
    public String getName() { return name; }

    new *
    public void setName(String name) { this.name = name; }
}
```

- ❖ Tạo file **item\_spinner.xml** để custom item Spinner có giao diện như đề bài.



- ❖ Tạo class **SchoolSpinnerAdapter** kế thừa class **BaseAdapter**, trong class **SchoolSpinnerAdapter** tạo các thuộc tính và constructor như sau:

```
public class SchoolSpinnerAdapter extends BaseAdapter {
    2 usages
    private final Context context;
    5 usages
    private final ArrayList<School> list;

    1 usage   ± Dinh Nguyen *
    public SchoolSpinnerAdapter(Context context, ArrayList<School> list) {
        this.context = context;
        this.list = list;
    }
}
```

- Xử lý các hàm đã implement

```
@Override
public int getCount() { return list.size(); }

± Dinh Nguyen
@Override
public Object getItem(int i) { return list.get(i); }

± Dinh Nguyen
@Override
public long getItemId(int i) { return i; }

± Dinh Nguyen *
@Override
public View getView(int i, View view, ViewGroup viewGroup) {
    LayoutInflater inflater = ((Activity) context).getLayoutInflater();
    view = inflater.inflate(R.layout.item_spinner, viewGroup, attachToRoot: false);

    ImageView ivLogo = view.findViewById(R.id.ivLogo);
    TextView txtName = view.findViewById(R.id.txtName);

    ivLogo.setImageResource(list.get(i).getImage());
    txtName.setText("FPoly " + list.get(i).getName());

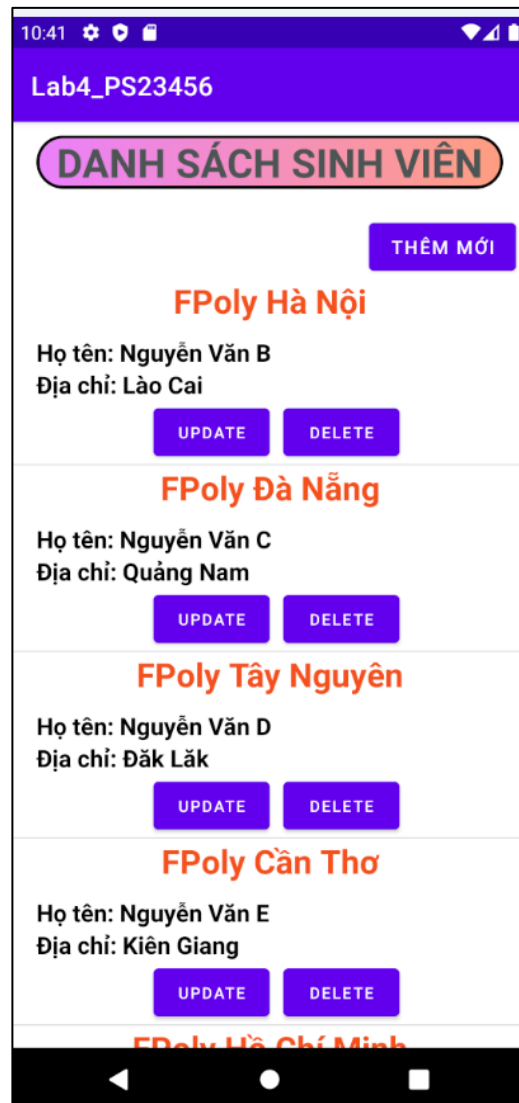
    return view;
}
```

❖ Trong **AddStudentActivity**, thêm một số dữ liệu cơ sở vào **danh sách**:

```
ArrayList<School> list = new ArrayList<>();  
list.add(new School(R.drawable.ha_noi, name: "Hà Nội"));  
list.add(new School(R.drawable.da_nang, name: "Đà Nẵng"));  
list.add(new School(R.drawable.tay_nguyen, name: "Tây Nguyên"));  
list.add(new School(R.drawable.tp_hcm, name: "Hồ Chí Minh"));  
list.add(new School(R.drawable.can_tho, name: "Cần Thơ"));
```

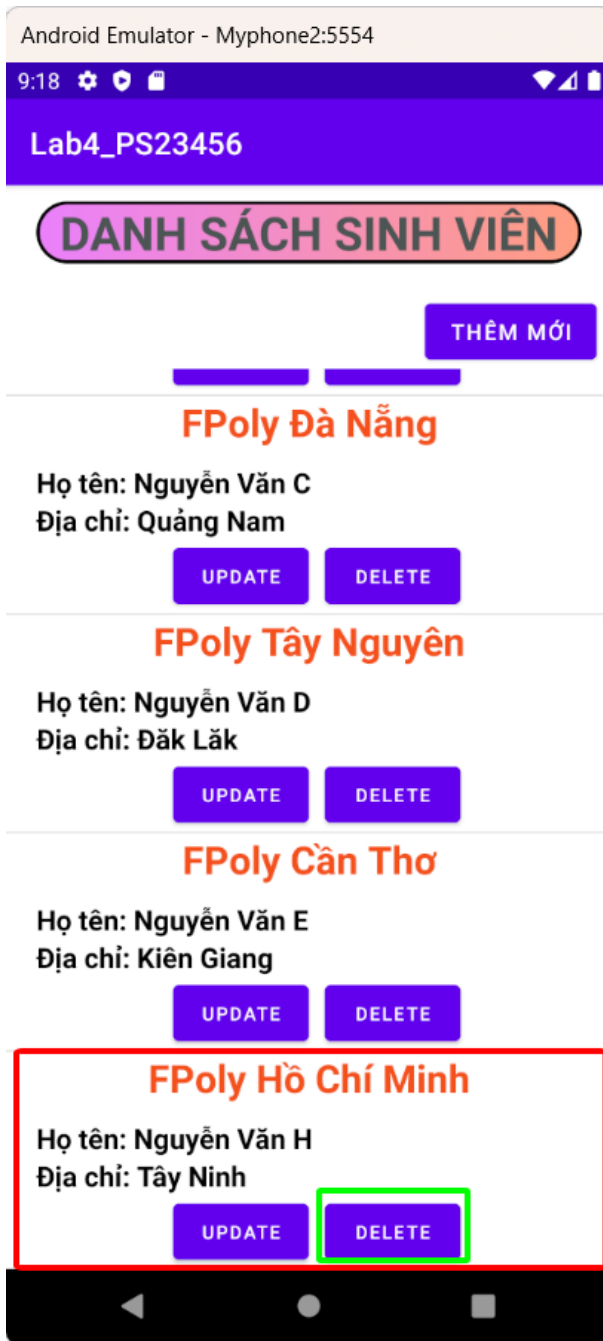
❖ Gán **SchoolSpinnerAdapter** cho **School Spinner**:

```
SchoolSpinnerAdapter schoolSpinnerAdapter = new SchoolSpinnerAdapter(context: this, list);  
spnSchool.setAdapter(schoolSpinnerAdapter);
```

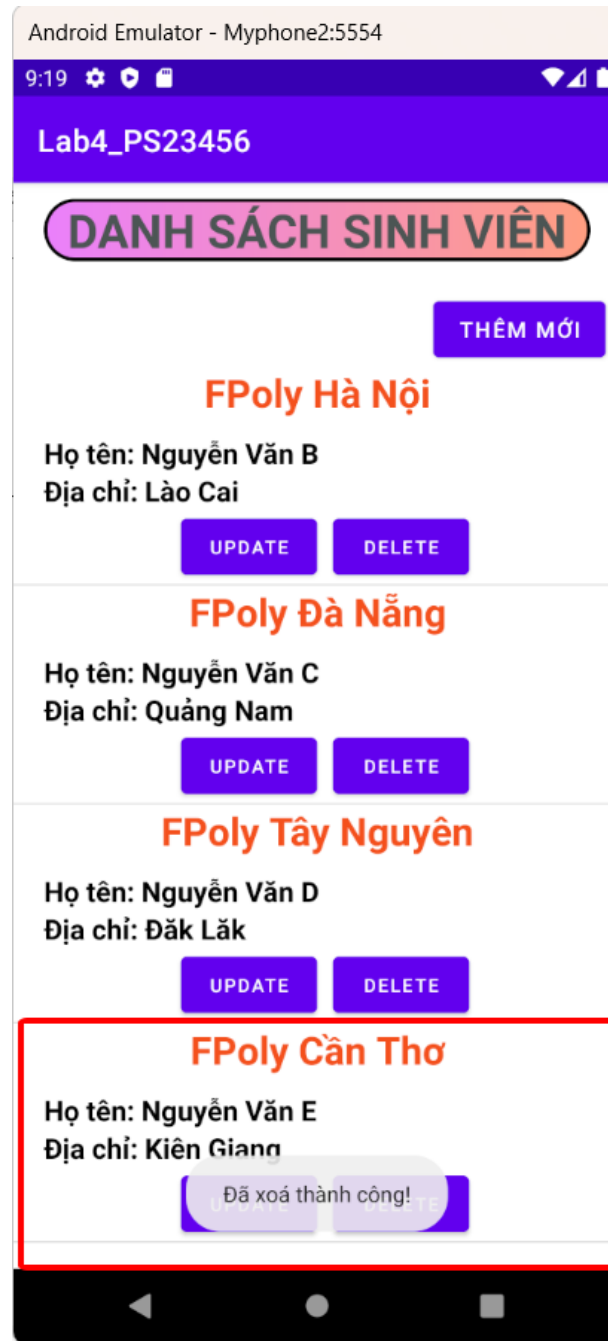
**BÀI 2: Tạo activity Home, Custom ListView và thiết kế giao diện như sau:****Hướng dẫn:**

- ❖ Tạo class **Student** gồm các thuộc tính: **branch, name, address**
- ❖ Tạo file **item\_listview.xml** để custom listview có giao diện như đề bài.
- ❖ Tạo adapter **ListStudentApdater** xử lý tương tự như làm với **SpinnerSchool**
- ❖ Trong activity **Home**, khởi tạo **ListStudentApdater** và gán adapter cho **ListView** (*sinh viên tự thêm dữ liệu vào list*)

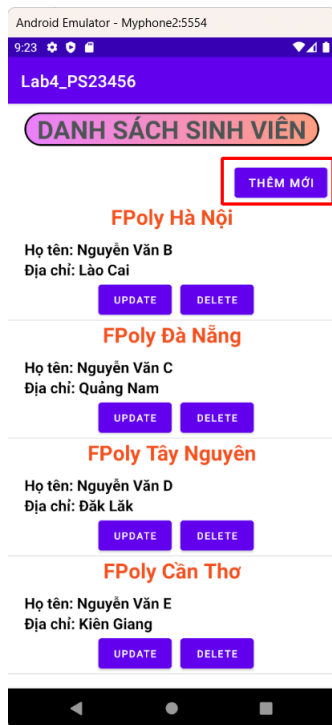
**BÀI 3: Kết hợp bài 1 và bài 2, thêm chức năng thêm mới và xoá item trong ListView, theo mô tả dưới đây:**



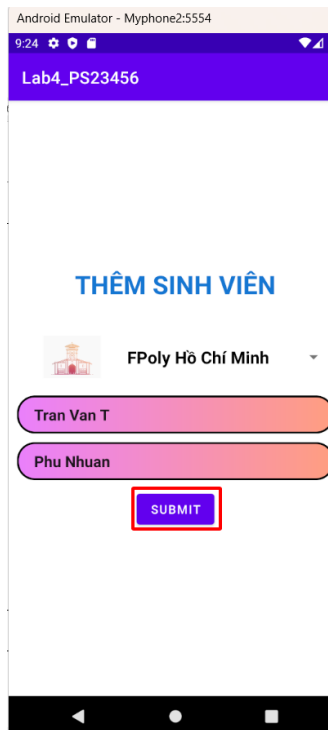
Trước khi click button **DELETE**



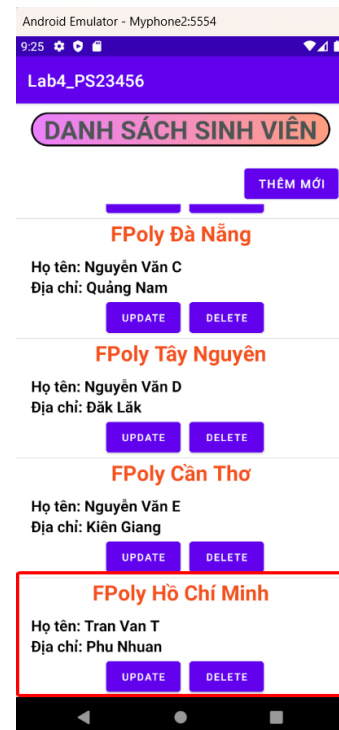
Sau khi click button **DELETE**



Nhấn **THÊM MỚI**



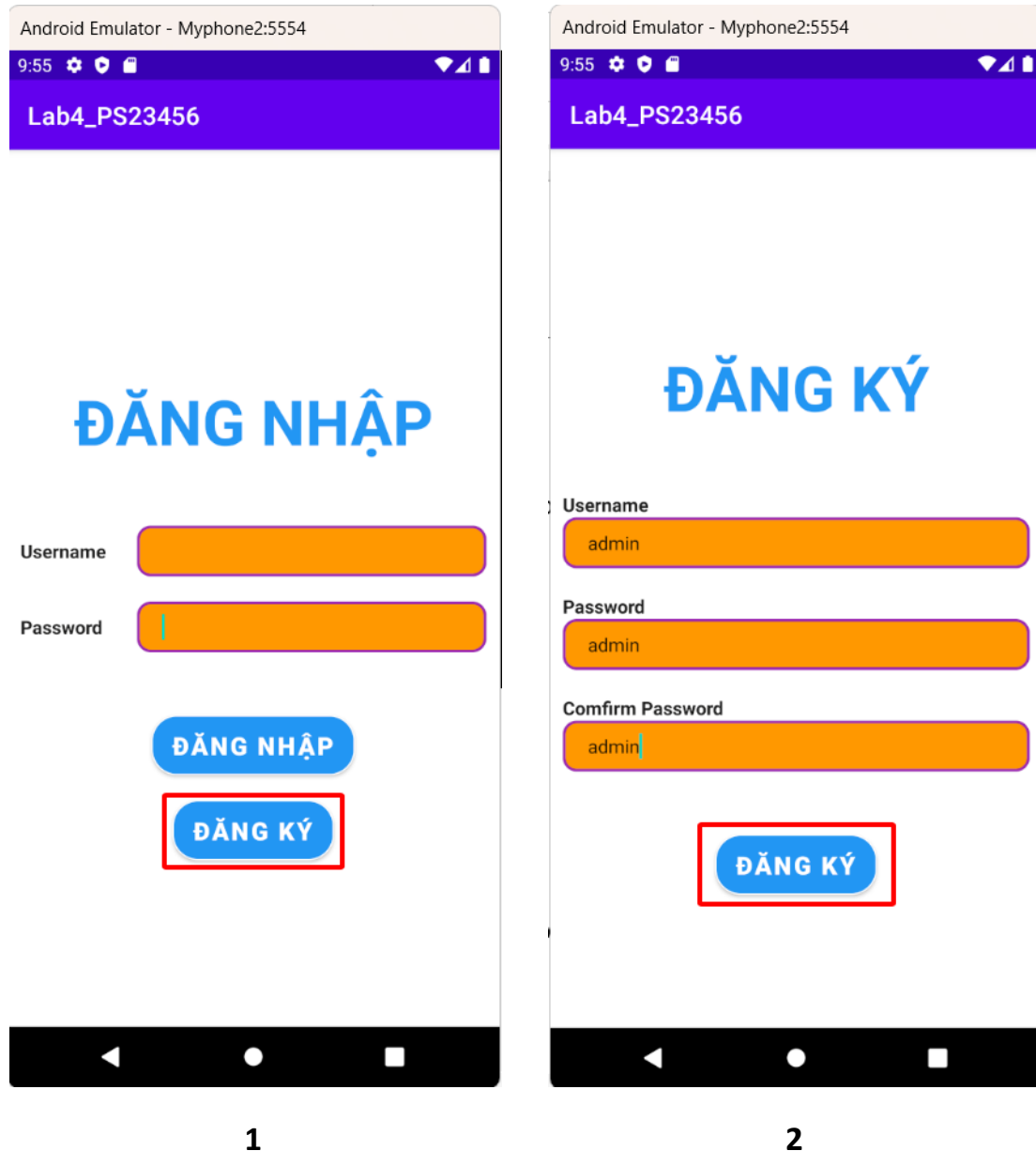
Điền thông tin và nhấn **SUBMIT**



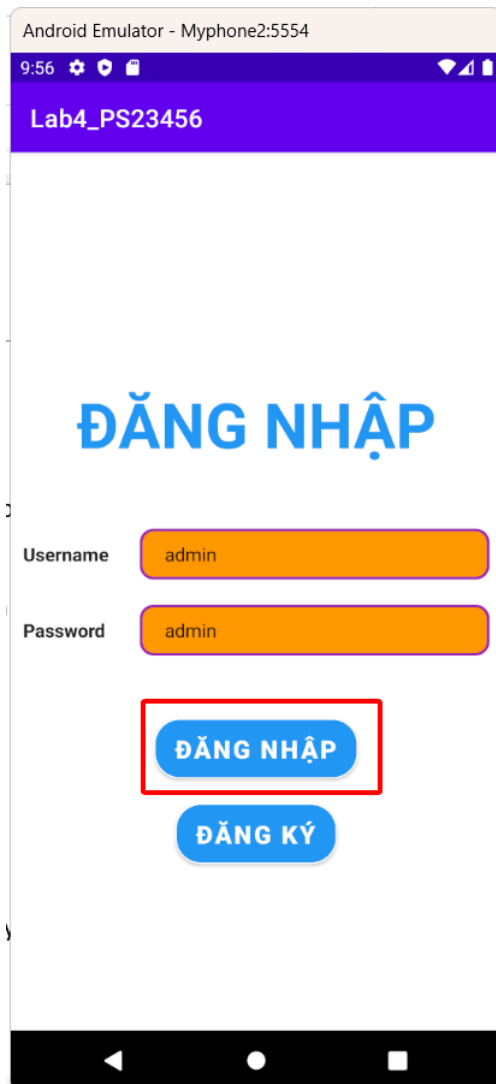
Sau khi nhấn **SUBMIT** tự động quay lại activity **Home** và đã xuất hiện **item mới**

**BÀI 4: Thực hiện thêm chức năng cập nhật thông tin sinh viên và đăng ký tài khoản**

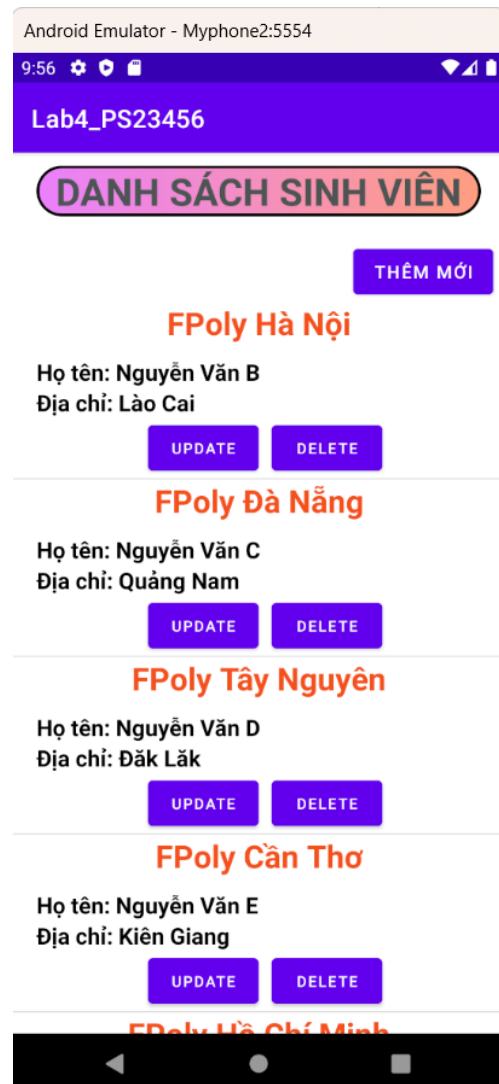
- ✓ Trước khi đăng nhập cần đăng ký tài khoản, sau khi đăng ký xong tiến hành đăng nhập và chuyển đến activity **Home**, theo thứ tự và minh họa dưới đây:







3



4

### BÀI 5: GV CHO THÊM

#### \*\*\* YÊU CẦU NỘP BÀI:

Sv nén file bao gồm các yêu cầu đã thực hiện trên, nộp lms đúng thời gian quy định của giảng viên. Không nộp bài coi như không có điểm.

--- Hết ---